

## Étapes détaillées du déploiement sur Render

Maintenant que notre API fonctionne en local, nous allons la **mettre en ligne gratuitement** grâce à la plateforme **Render**. Suivez pas à pas les instructions ci-dessous, même si vous n'avez jamais fait de déploiement auparavant !

---

### Étape 1 – Préparer les fichiers pour le Cloud

Avant d'envoyer notre code sur Render, nous devons vérifier deux choses :

#### 1.1 – Créer un fichier `requirements.txt` dans le dossier `api/`

Ce fichier liste les bibliothèques nécessaires pour faire tourner votre application sur Render. Voici les **lignes minimales** que vous devez y mettre :

```
fastapi>=0.115.12
httpx>=0.28.1
pydantic>=2.11.3
SQLAlchemy>=2.0.40
uvicorn>=0.34.0
```

Conseil : Ajoutez aussi d'autres dépendances si votre API en utilise (par exemple `uvicorn`, `pydantic`, etc.).

#### 1.2 – Pousser les modifications sur GitHub

Assurez-vous que tout votre code est bien **poussé (push)** sur GitHub. Render va se connecter à votre dépôt GitHub pour récupérer le projet.

---

### Étape 2 – S'inscrire sur Render

1. Allez sur <https://render.com>
2. Cliquez sur **"Sign up"** en haut à droite
3. Vous pouvez vous inscrire avec votre **compte GitHub** ou **Google** (très simple et rapide)
4. Une fois connecté(e), vous arrivez sur le **tableau de bord** Render

Aucun besoin de carte bancaire pour Render !

---

### Étape 3 – Créer un nouveau service Web

1. Cliquez sur le bouton **"New"** puis sélectionnez **"Web Service"**
- 

### Étape 4 – Connecter votre dépôt GitHub

1. Sélectionnez **"Public Git Repository"**
2. Collez l'**URL de votre dépôt GitHub** (celui contenant votre API)
3. Cliquez sur **"Connect"**

---

## Étape 5 – Configurer le déploiement

Sur la page suivante (où il est écrit "You are deploying a Web Service"), remplissez les champs comme ceci :

Champ	Valeur attendue
Name	Choisissez un nom unique (ex : <code>movielens-api</code> )
Project	Ne rien sélectionner / ne pas créer de projet
Language	<code>Python 3</code>
Branch	<code>main</code> (ou la branche où se trouve votre code)
Region	Choisissez la région la plus proche de chez vous
Root Directory	<code>api</code> (ou le dossier racine contenant <code>main.py</code> )
Build Command	<code>pip install -r requirements.txt</code>
Start Command	<code>uvicorn main:app --host 0.0.0.0 --port \$PORT</code>
Instance Type	<code>Free</code>

Le **Start Command** utilise `uvicorn` directement, sans `fastapi run`. C'est normal pour Render.

---

## Étape 6 – Lancer le déploiement

1. Descendez tout en bas de la page
2. Cliquez sur **"Deploy Web Service"**

Render commence à installer les dépendances, à démarrer votre API, et affiche les logs de déploiement.

---

## Étape 7 – Accéder à votre API en ligne

Quand le message **"Your service is live"** apparaît :

1. Copiez l'URL affichée en haut de la page (ex : `https://movielens-api-rmr7.onrender.com`)
2. Collez-la dans votre navigateur

Vous devriez voir le **message de santé** de votre API (si vous en avez défini un), ou bien la page de Swagger si vous allez sur `/docs` :

Exemple :

`https://movielens-api-rmr7.onrender.com/docs`

Cela ouvre l'interface **Swagger UI** pour tester vos endpoints.

## Étape 8 – Tester les endpoints de l'API via Swagger ou navigateur

Une fois l'API déployée, tu peux **tester tous les endpoints** comme tu l'avais fait en local. Il te suffit de te rendre à l'adresse suivante :

### Swagger UI :

<https://movielens-api-rmr7.onrender.com/docs>

Tu auras une interface interactive qui permet de tester facilement chaque route de l'API.

Tu peux aussi copier-coller l'URL dans ton navigateur pour faire les tests directement.

---

### Exemple 1 – Endpoint `/movies`

#### Lister tous les films :

<https://movielens-api-rmr7.onrender.com/movies>

#### Rechercher un film spécifique (ex: `movieId=1`) :

<https://movielens-api-rmr7.onrender.com/movies/1>

---

### Exemple 2 – Endpoint `/ratings`

#### Lister toutes les évaluations :

<https://movielens-api-rmr7.onrender.com/ratings>

#### Lister les évaluations pour un film donné (ex: `movieId=1`) :

[https://movielens-api-rmr7.onrender.com/ratings?movie\\_id=1](https://movielens-api-rmr7.onrender.com/ratings?movie_id=1)

---

### Exemple 3 – Endpoint `/tags`

#### Lister tous les tags :

<https://movielens-api-rmr7.onrender.com/tags>

#### Lister les tags pour un film donné (ex: `movieId=1`) :

[https://movielens-api-rmr7.onrender.com/tags?movie\\_id=1](https://movielens-api-rmr7.onrender.com/tags?movie_id=1)

---

### Exemple 4 – Endpoint `/links`

#### Lister tous les liens :

<https://movielens-api-rmr7.onrender.com/links>

#### Afficher le lien pour un film donné (ex: `movieId=1`) :

<https://movielens-api-rmr7.onrender.com/links/1>

---

### Exemple 5 – Endpoint `/analytics`

**Afficher les statistiques globales de la base de données :**

<https://movielens-api-rmr7.onrender.com/analytics>

Ce endpoint retourne un résumé contenant :

- le nombre total de films
  - le nombre total d'évaluations
  - le nombre total de tags
  - le nombre total de liens
-